In a Pro plan project, you notice there is a mail delivery problem in the Integration environment.
How do you troubleshoot this?

☑ **A**
Move the test to the Staging environment

☐ **B**
Update the SPF record in the DNS

☐ **C**
Look in the var/log directory for the mail.log file

☐ **D**
Change the log level in the log:email section of the .magento.env.yaml file
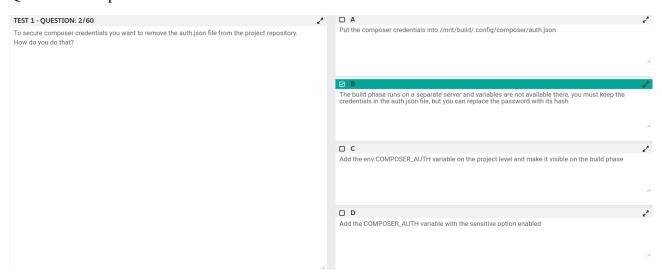
Question #2 Topic 1

You have a Starter project. Your code is ready for testing in the Staging environment, but you need the latest data from Production.
What do you do to update the Staging environment?

- A. Log in to the Project Web UI, choose the Staging environment, and click Sync
- **B. Log in to the Project Web UI, choose the Staging environment, and click Merge**
- C. Checkout the Staging branch and use the git push origin -f command
- D. Checkout the Production environment and use the magento-cloud sync CLI command

Question #3 Topic 1

To secure composer credentials you want to remove the auth.json file from the project repository.
How do you do that?

☐ **A**
Put the composer credentials into /mnt/build/.config/composer/auth.json

☑ **B**
The build phase runs on a separate server and variables are not available there, you must keep the credentials in the auth.json file, but you can replace the password with its hash

☐ **C**
Add the env:COMPOSER_AUTH variable on the project level and make it visible on the build phase

☐ **D**
Add the COMPOSER_AUTH variable with the sensitive option enabled

Question #4 Topic 1

While investigating an inherited Magento Commerce Cloud project, you notice the following
SCD_MATRIX configuration in .magento.env.yaml: stage: global:
SCD_MATRIX:
"magento/backend":
language:

- en_US

What is the effect of this configuration?
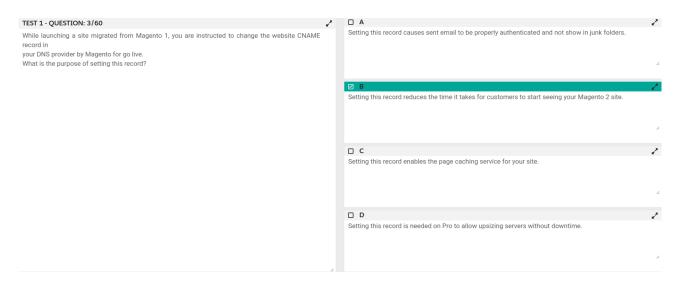
- A. Static content for frontend themes is generated during the deploy phase
- B. All frontend theme static content will be reused from the previous deployment

While launching a site migrated from Magento 1, you are instructed to change the website CNAME record in
your DNS provider by Magento for go live.
What is the purpose of setting this record?

A

Setting this record causes sent email to be properly authenticated and not show in junk folders.

☑ B

Setting this record reduces the time it takes for customers to start seeing your Magento 2 site.

C

Setting this record enables the page caching service for your site.

D

Setting this record is needed on Pro to allow upsizing servers without downtime.

Question #6 Topic 1

While launching a site migrated from Magento 1, you are instructed to change the website CNAME record in your DNS provider by Magento for go live.
What is the purpose of setting this record?

- A. Setting this record causes sent email to be properly authenticated and not show in junk folders.
- B. Setting this record enables the page caching service for your site.
- C. Setting this record is needed on Pro to allow upsizing servers without downtime.
- **D. Setting this record reduces the time it takes for customers to start seeing your Magento 2 site.**

You need to directly access the database that belongs to your Integration environment from your local environment.
How do you achieve this?

A

Use the magento-cloud db:sql command

B

Find the connection information from the env.php file and connect with your local MySQL client

☑ C

Add the ece-tools db-expose command to your deploy hook

D

Collect the SSH connection details using magento-cloud ssh --pipe and connect with your local MySQL client

A client has an in-house design team which manages content on their Magento Commerce Cloud project.
This content is rapidly changing and they would like the ability to maintain their own stylesheets in a structured way without affecting site uptime.
How do you achieve this on the Integration branch?

☐ A
Modify the writeable directories on the Integration Environment by using the .magento.env.yaml file.

☑ B
Use the .magento.app.yaml file to create a mount in the pub directory.

☐ C
The Magento Infrastructure Team needs to create a writable directory in which the stylesheets can be uploaded.

☐ D
Add a custom command to the build hook in the .magento.app.yaml file to retrieve the stylesheets from a client hosted location.

Question #9 Topic 1

A client has an in-house design team which manages content on their Magento Commerce Cloud project. This content is rapidly changing and they would like the ability to maintain their own stylesheets in a structured way without affecting site uptime.
How do you achieve this on the Integration branch?

- A. Modify the writeable directories on the Integration Environment by using the .magento.env.yaml file.
- B. Add a custom command to the build hook in the .magento.app.yaml file to retrieve the stylesheets from a client hosted location.
- C. The Magento Infrastructure Team needs to create a writable directory in which the

You want to exclude some themes from static assets generation only for integration environments. You are
trying to choose where to set the SCD_EXCLUDE_THEMES variable in the .magento.env.yaml file or in the Project Web UI.
What prevents the usage of the .magento.env.yaml file?

☐ A
The code from integration would be merged into the Staging and Production branches

☐ B
Child environments inherit the variable when you use the .magento.env.yaml file

☑ C
The .magento.env.yaml file can contain only deployment configuration options, which are only available if static assets generation runs on the deploy phase

☐ D
Only variables which are set using the Project Web UI has the option visible during build

Question #11 Topic 1

You are migrating an existing Magento 2 website to Magento Commerce Cloud. After setting up the project you would like to migrate the database to the
Integration branch. The database import fails with a disk space error.
How do you migrate the database without data loss?
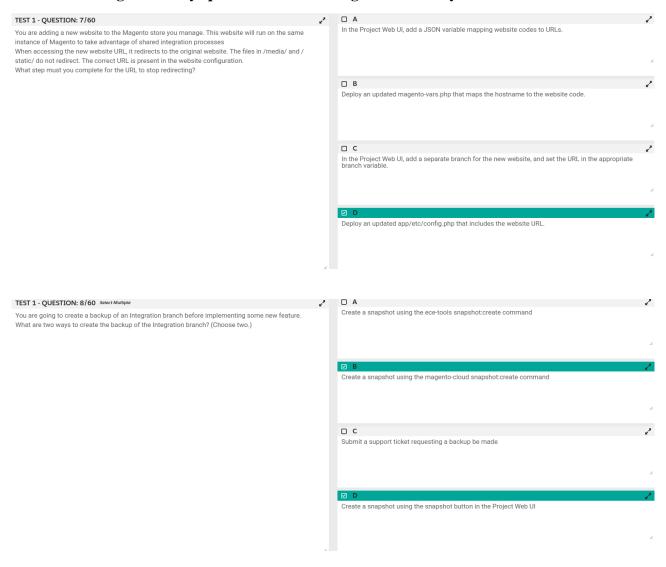
- A. Configure the mysql disk size using a variable in the .magento.env.yaml file
- B. Open a Magento Support ticket to increase the Integration branch disk size

- C. Trim down the database to a smaller size
- **D. Configure the mysql disk size in the .magento/services.yaml file**

You are adding a new website to the Magento store you manage. This website will run on the same instance of Magento to take advantage of shared integration processes
When accessing the new website URL, it redirects to the original website. The files in /media/ and /static/ do not redirect. The correct URL is present in the website configuration.
What step must you complete for the URL to stop redirecting?

☐ **A**
In the Project Web UI, add a JSON variable mapping website codes to URLs.

☐ **B**
Deploy an updated magento-vars.php that maps the hostname to the website code.

☐ **C**
In the Project Web UI, add a separate branch for the new website, and set the URL in the appropriate branch variable.

☑ **D**
Deploy an updated app/etc/config.php that includes the website URL.

You are going to create a backup of an Integration branch before implementing some new feature.
What are two ways to create the backup of the Integration branch? (Choose two.)

☐ **A**
Create a snapshot using the ece-tools snapshot:create command

☑ **B**
Create a snapshot using the magento-cloud snapshot:create command

☐ **C**
Submit a support ticket requesting a backup be made

☑ **D**
Create a snapshot using the snapshot button in the Project Web UI

Question #14 Topic 1

You are a technical admin on a Magento Commerce Cloud account. A coworker needs to submit a support request with Magento Support.
What is needed to enable support ticket access?

- **A. Ask the account owner to add the user to shared access.**
- B. Ask the launch manager to create a support account for this user.
- C. Change the environment access role to contributor for this user.
- D. Update the project role of this user to administrator.

Question #15 Topic 1

For the Magento search backend, you would like to move from MySQL to Elasticsearch for your existing
project. You have confirmed that your services.yaml file is configured appropriately. However, after your
most recently deployment, you notice the search engine is still set to MySQL.
What additional step must be completed?

☑ **A**

Remove the MySQL service from the services.yaml file

☐ **B**

Find an appropriate Elasticsearch module to require with Composer

☐ **C**

Specify Elasticsearch as the search engine in the Magento Admin panel

☐ **D**

Configure the relationships section of the .magento.app.yaml file

Question #16 Topic 1

On a project that deploys static content during the build phase, a merchant states the deploy phase is still taking too long. You consider turning off JavaScript minification to reduce the build time. Besides reducing the build phase time, what two consequences does turning off JavaScript minification have? (Choose two.)

- **A. The deploy artifact size will be decreased because of the larger JavaScript can be symlinked**
- B. Browsing the store will be slower because larger JavaScript files have to be downloaded
- C. The deploy phase will be shorter because JavaScript can be symlinked from init instead of copied

You are migrating an existing Magento 2 website to Magento Commerce Cloud. After setting up the project
you would like to migrate the database to the Integration branch. The database import fails with a disk
space error.
How do you migrate the database without data loss?

☐ **A**

Configure the mysql disk size using a variable in the .magento.env.yaml file

☑ **B**

Configure the mysql disk size in the .magento/services.yaml file

☐ **C**

Open a Magento Support ticket to increase the Integration branch disk size

☐ **D**

Trim down the database to a smaller size

You need to increase the Admin Session Lifetime up to 86400 seconds for the branch named Integration
and children branches, but not for the Staging and Production branches.

How do you update this on the Integration branch and its child branches?

☑ A

On the Integration environment, set the Admin Session Lifetime in the Magento admin backend under Stores > Configuration > Advanced > Admin > Security

☐ B

Run the following Magento CLI command:
php bin/magento config:set admin/security/session_lifetime 86400

☐ C

On your local environment, set the Admin Session Lifetime in the Magento admin backend under Stores > Configuration > Advanced > Admin > Security
then dump this setting into the app/etc/config.php file and commit this file

☐ D

Add the following variable on the Integration branch and set its value to 86400:
env:CONFIG_STORES_DEFAULT_ADMIN_SECURITY_SESSION_LIFETIME

Question #19 Topic 1

Magento Support advises you to upgrade to the latest release of ece-tools matching the project's Magento Commerce version 2.3.1.
How do you do that?

- A. Require the exact version with composer require magento/ece-tools:2.3.1-pX where X is the latest release number
- B. Clone the repository github.com/magento/ece-tools and copy the src/ folder to vendor/magento/ece-tools/src
- **C. Run the command composer update magento/ece-tools**
- D. Run the command ece-tools self-upgrade

On a project that deploys static content during the build phase, a merchant states the deploy phase is still
taking too long. You consider turning off JavaScript minification to reduce the build time.

Besides reducing the build phase time, what two consequences does turning off JavaScript minification
have? (Choose two.)

☐ A

The deploy phase will be shorter because JavaScript can be symlinked from init instead of copied

☐ B

Browsing the store will be slower because larger JavaScript files have to be downloaded

☑ C

The deploy artifact size will be decreased because of the larger JavaScript can be symlinked

☑ D

The build phase will be longer because the additional pass of JavaScript merging

Question #21 Topic 1

You are going to create a backup of an Integration branch before implementing some new feature.
What are two ways to create the backup of the Integration branch? (Choose two.)

- A. Create a snapshot using the ece-tools snapshot:create command
- **B. Create a snapshot using the snapshot button in the Project Web UI**
- **C. Create a snapshot using the magento-cloud snapshot:create command**

- D. Submit a support ticket requesting a backup be made

Question #22 Topic 1

A merchant using Magento Commerce Cloud Pro reports an issue with an order missing transaction

TEST 1 - QUESTION: 13/60

While investigating an inherited Magento Commerce Cloud project, you notice the following SCD_MATRIX

configuration in .magento.env.yaml:
stage:
  global:
    SCD_MATRIX:
      "magento/backend":
        language:
          - en_US
What is the effect of this configuration?

☑ A
Static content for all locales will be generated for all frontend themes

☐ B
Static content for frontend themes is generated during the deploy phase

☐ C
All frontend theme static content will be reused from the previous deployment

☐ D
Only the static content for the adminhtml area will be generated

TEST 1 - QUESTION: 14/60

You are testing an integration to an external service that requires you to configure an access key in an environment variable. You created a new environment to validate the functionality. The variable must only
impact this environment.
How do you configure the variable?

☐ A
Use vendor/bin/ece-tools to generate a local config.php file and commit the setting

☐ B
Add the variable to the variables section under Project Settings specifying inheritance level

☐ C
Use the magento-cloud CLI utility with the environment, inheritable and level flags

☑ D
Add the variable to the .magento.env.yaml file specifying the environment and inheritable

Question #25 Topic 1

You are reviewing a third party extension for Magento Commerce Cloud compatibility.
Which pattern makes a module incompatible with Magento Commerce Cloud?

- A. Direct instantiation by the ObjectManager of non-generated classes in .phtml template files
- **B. Direct instantiation by the ObjectManager in non-constructor methods with the class also being referenced in the constructor signature**
- C. Configuration referencing generated Proxy classes in the modules etc/frontend/di.xml file
- D. Direct instantiation of generated classes by the ObjectManager without the class being referenced in the constructor

You have a Starter project. Your code is ready for testing in the Staging environment, but you need the latest data from Production.
What do you do to update the Staging environment?

☐ A
Checkout the Production environment and use the magento-cloud sync CLI command

☐ B
Log in to the Project Web UI, choose the Staging environment, and click Sync

☑ C
Log in to the Project Web UI, choose the Staging environment, and click Merge

☐ D
Checkout the Staging branch and use the git push origin -f command

Question #27 Topic 1

In a Pro plan project, you notice there is a mail delivery problem in the Integration environment. How do you troubleshoot this?

- A. Change the log level in the log:email section of the .magento.env.yaml file
- B. Update the SPF record in the DNS
- C. Look in the var/log directory for the mail.log file
- **D. Move the test to the Staging environment**

Question #28 Topic 1

You are reviewing a third party extension for Magento Commerce Cloud compatibility. You

You are upgrading a project to the latest version of Magento Commerce Cloud and part of the process involves a PHP version upgrade. This is now ready to be tested by the QA Team on the Integration environment.
How do you apply the PHP version upgrade?

☐ A
Change the PHP_VERSION configuration in the .magento.env.yaml file and re-deploy

☑ B
Change the PHP version in the .magento.app.yaml file and re-deploy

☐ C
Use the magento-cloud CLI tool to update the PHP_VERSION variable

☐ D
SSH into the Integration environment and upgrade PHP manually

A custom module that adds a new console command for bin/magento has just been built. Deploying this
to the Production environment on your Pro plan site causes the build phase to fail.
You find this occurs when you inject certain Magento core classes in your console command class. The
command works on your local environment.
Why does this occur?

A
A connection to the database was attempted, which is not available during the build phase.

B
The injected class logged to the exception log file, but a different logging implementation is used during
deployment.

C
The three webservers generated different content, which causes the build to be rolled back for safety.

D
Files were written to the root Magento directory, which is read-only on Magento Commerce Cloud.

A merchant states the downtime during the deployment phase is too long. After analysis, it is determined
the static content deployment is the longest process. Static content needs to be available immediately after
deployment.
How can the static content deployment time be reduced?

A
Commit var/view_preprocessed to git so it already exists during the deployment phase

B
Reduce the number of store views per website

C
Set stage/global/SKIP_SCD to true

D
Move static content deploy to the build phase

You are reviewing a third party extension for Magento Commerce Cloud compatibility.
Which pattern makes a module incompatible with Magento Commerce Cloud?

A
Direct instantiation by the ObjectManager in non-constructor methods with the class also being referenced in the constructor signature

B
Direct instantiation of generated classes by the ObjectManager without the class being referenced in the constructor

C
Direct instantiation by the ObjectManager of non-generated classes in .phtml template files

D
Configuration referencing generated Proxy classes in the modules etc/frontend/di.xml file

A merchant using Magento Commerce Cloud Pro reports an issue with an order missing transaction data.
The application uses a payment gateway integration which posts a series of callbacks to Magento.
You would like to retrieve a complete list of calls to build a timeline of what happened.
How do you achieve this?

A
Use SSH to access all nodes and investigate the access.log files

B
View the access.log file in the Project Web UI

C
Use the magento-cloud environment:ssh command to access the environment and investigate the access.log file

D
Use the magento-cloud environment:logs command to retrieve the access.log file from Production

You want to improve the ability to monitor Production deployments by setting up an email notification system.
How do you achieve this?

A

Build a custom module which hooks into the deployment phase and sends emails in real-time

B

Enable Deployment Notifications under Configure Environment > Settings in the Project Web UI

☑ C

Configure log notifications in the .magento.env.yaml file

D

Use the magento-cloud environment:deploy:email command to enable email notifications

Your company decided to move a Magento store to a subdomain https://shop.company.com/ and redirect traffic from the old URL https://company.com/ to the new one. Your company is using the Starter plan.
How do you implement the redirect?

☑ A

Add a route into the .magento/routes.yaml configuration file.

B

Add a route using the MAGENTO_CLOUD_ROUTES environment variable.

C

Configure a rewrite rule in the .htaccess file in the pub folder.

D

Submit a support ticket to add the redirect on the web-server level.

You have created a new Integration branch and did not receive the administrator email with the password
reset link.
What is prohibiting the email from sending?

☐ **A**

Email is always disabled on all integration branches

☐ **B**

You have not enabled the Magento Email module with bin/magento module:enable

☐ **C**

You have not configured SendGrid for this environment

☑ **D**

The Outgoing Emails setting is disabled under Environment Settings in the Project Web UI

You are debugging an issue with an extension that only occurs in Production. The extension writes relevant
information using the debug log level.
However, that specific information is not getting written to the logs on Production.
How do you fix this?

A

Enable debug logging in the Admin interface in Production

B

Write a custom logger that writes the debug information to var/log/custom.log

C

Enable developer mode in Production for a short period to gather the required information

D

Enable debug logging in app/etc/config.php locally and deploy it to Production

Your team is working on two major changes affecting the product page and you want the ability to
implement and test these changes in parallel. Your company is using the Pro plan
How do you achieve this using the Project Web UI?

A

Create branches from Production to ensure the catalog is up to date

B

Create branches from Integration to facilitate merging

C

Create branches from Staging to ensure the changes are tested on an environment similar to
Production

D

Create branches from the Magento Cloud Docker image to ensure the changes are made on the most
recent codebase

You are a technical admin on a Magento Commerce Cloud account. A coworker needs to submit a support
request with Magento Support.
What is needed to enable support ticket access?

A

Ask the account owner to add the user to shared access.

B

Change the environment access role to contributor for this user.

C

Ask the launch manager to create a support account for this user.

D

Update the project role of this user to administrator.

You fixed an issue in a third-party extension using a patch named m2-hotfixes/extension-fix.patch.
Before committing the patch to the repository you want to test it locally.
How do you test a patch in a clean working copy?

A

Run the command bin/magento setup:hotfix:apply --patch=m2-hotfixes/extension-fix.patch

B

Commit the patch because the pre-commit hook will abort the commit if any hotfix causes an issue

C

Configure a deploy hook in .magento.app.yaml that executes vendor/bin/ece-tools hotfix:apply --dry-run

D

Run the command git apply m2-hotfixes/extension-fix.patch

You upgraded an integration branch in your Magento Commerce Cloud project, but received an error during
the deploy phase.
What are the two ways to review details about the deployment? (Choose two.)

☑ **A**

View the logs in the var/reports/ directory on the remote server

☑ **B**

View the deploy.log file in the /var/log/ directory

☐ **C**

View the cloud.log file in the var/log/ directory

☐ **D**

View the exception.log file in the var/log/ directory

You cloned the Integration branch to your local environment and imported the database dump from Integration. You performed composer install and bin/magento setup:install.
While placing an order using PayPal Express, the following error occurs:
PayPal gateway has rejected request. Security header is not valid (#10002:
Security error).
What is the cause of this error?

☑ **A**

PayPal Sandbox API credentials are not valid for the local environment.

☐ **B**

The PHP extension mcrypt has not been installed locally.

☐ **C**

A new encryption key has been created on the setup:install action.

☐ **D**

The folder var/session has no write permissions for the web server user.

You added a grunt autoprefixer command, which adds CSS vendor prefixes like -webkit- and -

moz- to CSS files generated by the setup:static-content:deploy command.
After deployment you still see CSS files without prefixes. The hooks section in the .magento.app.yaml file is:
hooks:
  build: |
    npm install
    grunt autoprefixer
    php ./vendor/bin/ece-tools build:generate
    php ./vendor/bin/ece-tools build:transfer
deploy: |
    php ./vendor/bin/ece-tools deploy
Considering static assets are being generated on the build phase, why are CSS prefixes missing?

☐ **A**

The static assets were not generated yet when the grunt command ran

☐ **B**

Custom commands can be run only on the deploy phase

☐ **C**

The custom command was run before static assets were transferred into the init directory

☑ **D**

CSS vendor prefixes must be added to CSS files locally and committed as part of a theme

After making changes to custom commands in the deploy and build phases, you deploy to Staging. The
deployment takes much longer than expected, and Staging is in maintenance mode for most of this time.
What do you do to reduce site downtime?

☐ A

Move custom commands from the deploy phase to the build phase.

☑ B

Set the SKIP_MAINTENANCE_SCD variable to true in the .magento.env.yaml file.

☐ C

Add a custom command to the post_deploy phase to activate deferred deployment.

☐ D

Update the SCD_MATRIX variable to include only necessary custom commands.

You built a custom command that runs during the build phase, but it is not generating files properly.
Magento Support provided a patch that you added to the project.
What must you do to ensure this command runs after the patch has been applied?

A

Move your custom command to run in the deploy phase.

B

Nothing, because patches are already applied when any hook runs.

C

Apply the patch manually in your custom command, because patches are always applied at the end of deployment.

☑ **D**

Ensure your custom command runs after the ece-tools build hooks.

You have a project deploying Static Content on the deploy phase. This process is optimized using the SCD_MATRIX configuration in the deploy stage of the .magento.env.yaml file.
After moving the process to the build phase, Static Content Deployment is no longer optimized and all themes and all locales are generating.
How do you resolve this?

A

Add the --use_scd_matrix flag to the ece-tools build:generate command in the .magento.app.yaml file

B

Configure the variable ENABLE_SCD_MATRIX in the .magento.env.yaml file

☑ **C**

Move the SCD_MATRIX configuration to the build stage of the .magento.env.yaml file

D

The SCD_MATRIX configuration is only used on the deploy phase

After deploying to Staging for the first time with the Fastly module installed, you notice that Fastly is not
caching pages. Page caching works properly in a local development environment.
What two steps are required to make Fastly cache pages? (Choose two.)

☐ **A**

Enable the Fastly connection by running magento-cloud fastly:setup.

☑ **B**

Activate the module by setting Caching Application to Fastly CDN.

☐ **C**

Connect the Fastly shield using the Shield setting in the Magento admin.

☑ **D**

Populate VCL in Fastly by clicking on Upload VCL to Fastly in the Magento admin.

You added the env:ADMIN_PASSWORD variable in the Project Web UI to change a Magento admin user's
password. After deployment you are unable to login using the new password.
What causes this?

A

Deploy scripts read configuration from the environment variable called $MAGENTO_CLOUD_VARIABLES,
which contains an array of variables which were set without the env: prefix

B

The sensitive option is required for env:ADMIN_PASSWORD variable

C

When you add a variable, the build stage is being skipped, because the codebase has not been
changed. You must push a commit to trigger a full deploy

☑ D

Variables which are set using the Project Web UI are not available on the build phase, the admin
password variable should be set in the .magento.env.yaml file

The merchant wants to offer specific pricing in different currency markets, so you have set up multiple
websites to cover each currency.
Now they want customers to be directed to the appropriate website based on the country they are in.
How do you configure this?

☑ A

Build or install a Magento extension that changes the default currency code on application init.

B

In the Project Web UI, enable the Website IP Mapping setting.

C

Use the Country Mapping in the Fastly settings to map country codes to store views.

D

Update magento-vars.php to detect the country code from the IP address and map it.

After installing a Magento extension that implements a new class for a site and deploying this to an integration environment, you see it is not working.

In the logs, you discover the following error:
The specified /app/generated/code/MyCompany/MyModule/MyClassFactory.php could not be written failed to open stream: Read-only file system
How do you fix this?

☐ **A**

A support ticket should be filed with Magento Commerce Cloud to fix the filesystem issue.

☐ **B**

The GENERATED_CODE_WRITABLE variable should be set to true whenever using extensions.

☐ **C**

The file should be generated locally and pushed via Git.

☑ **D**

The extension should be modified so it does not write to generated at runtime.

While setting up a Magento project, you run the following command:
magento-cloud variable:update ADMIN_USERNAME --value=alice
What will this command do?

☐ A

Configure the Magento Commerce Cloud tools to use alice for API call authentication.

☐ B

Change your username to alice for logging into the magento-cloud CLI tool.

☑ C

Update the Magento cloud/admin/username config variable to alice.

☐ D

Change the username of the default administrator account to alice.

You are setting up a third-party integration only for the Production environment, which connects to an external RabbitMQ server.
Where do you put the configuration?

☐ A

In the env:QUEUE_CONFIGURATION variable added in the Project Web UI or magento-cloud CLI

☐ B

In the QUEUE_CONFIGURATION variable in the .magento.env.yaml file

☐ C

In the app/etc/env.php file

☑ D

In the QUEUE_CONFIGURATION variable added in the Project Web UI or magento-cloud CLI

You need to get code with a backend order placement security related fix into production as quickly as possible.

The project currently deploys static content during the build phase with both stage/global/SKIP_SCD and stage/deploy/STATIC_CONTENT_SYMLINK set to false.

What setting in magento.env.yaml minimizes the overall deployment time?

A

Set stage/global/STATIC_CONTENT_SYMLINK to true to symlink static assets to the init directory

B

Set stage/global/SKIP_SCD to true to completely avoid static content processing

C

Set stage/global/SCD_COMPRESSION_LEVEL to 0 to avoid the compression overhead

D

Remove all themes from stage/global/SCD_MATRIX so they will not be generated

You want to move static content deploy to the build phase of deployments.
Which two actions do you take? (Choose two.)

A

Run ece-tools config:dump on production

B

Run ece-tools scd-deploy:set build on production

C

Use scp to copy app/etc/config.php from local to production

D

Download and commit app/etc/config.php from production

You need to install a third party extension which is provided in a tar archive. It expands to a folder structure
in app/code/MyCompany/MyModule.
Which action do you take to install the extension?

A

Use the composer-merge-plugin to enable support for extension in app/code

B

Request composer repository credentials from the extension vendor

C

Add it to the project in app/code and commit it to git

D

Use the composer package command to create a composer package and install it with composer require

You need to specify the admin password using an environment variable. You have created an environment
variable env:ADMIN_PASSWORD with a valid password. When attempting to log in to the Magento Admin it
is not accepting the new password.
How do you correct the environment variable?

☐ A

The ADMIN_PASSWORD variable should be configured via .magento.env.yaml

☐ B

The env:ADMIN_PASSWORD variable can only be used for an initial installation

☑ C

The environment variable should not have the env: prefix

☐ D

The Sensitive option is checked by default and must be disabled

A merchant with three websites using en_GB, fr_FR and de_AT requires a change in locale to de_DE for the German Production website. After running the following command on Production you notice the frontend

design of the German website is broken:

php bin/magento config:set --scope=websites --scope-code=german general/locale/

code de_DE

Static content deployment is set up on the deploy phase.

Why did this happen?

A

de_DE is not writeable in pub/static and has to be configured via the .magento.app.yaml file

B

STATIC_CONTENT_SYMLINK is not configured in the .magento.env.yaml file

☑ C

The magento-cloud environment:deploy command has not been run yet to generate static content for the new locale

D

Static Content is not pre-generated for the de_DE locale

You are updating a Magento Commerce Cloud project from version 2.2.5 to 2.3.0 with the command composer require magento/product-enterprise-edition:2.3.0

However, composer is unable to complete the task because the installed version of magento/magento-

cloud-metapackage is not compatible.

How do you upgrade Magento to the desired version?

A

Run composer require magento/magento-cloud-metapackage:">=2.3.0 <2.3.1" --update-with-dependencies

☑ B

Revert the composer.json to the last commit before the update and run the command again with the --with-dependencies argument

C

Remove magento/product-enterprise-edition from the root composer.json requirements and run composer update

D

Create a new project based on magento/project-enterprise-edition:2.3.0, then add ece-tools, custom modules and themes again with composer and migrate the data

You migrated a live On-Premise application to Magento Commerce Cloud Pro. The performance on Staging
and Production is fine. However, the internal testing team notices overall performance degradation on the
Integration environment.
Why is this happening?

A

The Staging and Production environments are consuming all available resources

B

The Integration branch services have not been optimized in the .magento/services.yaml file

C

The Integration branch uses Platform-as-a-Service shared resources

D

Xdebug is always enabled on the Integration branch

You are working with a new developer who needs to be able to deploy code and view logs on Magento Commerce Cloud.
What must you do to give them access?

☑ **A**

Add their SSH public key to a file named .ssh/authorized_keys and deploy the change.

☐ **B**

In the Cloud Project Web UI, click Add User to add their email. They need to enter their SSH public key.

☐ **C**

In the .magento.app.yaml file, add their email under Users. They need to enter their SSH public key.

☐ **D**

Add a Magento admin user for them and paste the SSH public key into their user's SSH Key field, which
is available only on Magento Commerce Cloud.

You created a custom module that is not functioning as expected on your Integration environment. You

would like to debug the code using Xdebug. Xdebug is missing from the output of the command php -m.
What are two ways to load the Xdebug module? (Choose two.)

☑ **A**

Add xdebug into the .magento.app.yaml file under the extensions section

☐ **B**

You log a support request as PHP modules can only be installed by support

☑ **C**

Create a custom php.ini that includes the extension

☐ **D**

In the phpserver folder, create a folder conf.d and add a file ext-xdebug.ini that includes the extension

You are tasked to improve the user experience of the most visited product pages after deploying releases
to Production. The aim is to reduce initial page load times.
How do you achieve this?

A

Modify the post_deploy hook in the .magento.app.yaml file to include the URLs as arguments to the ece-tools command

B

Use the WARM_UP_PAGES variable in the .magento.env.yaml file to cache the pages during post_deploy

C

Use the Magento Admin to configure Fastly to cache the pages after deployment

D

Create a custom build command and use CURL to cache the page

You need to test a data-related issue occurring on your Production environment. You replicate the data to
your staging environment utilizing the Synchronization tool. When you access your staging environment,
you are redirected to your Production environment. Your project is on the Starter plan.
How do you prevent the redirect?

☑ A

Enter the proper URL in the Routes for Staging under the Environment Settings in the Project Web UI

☐ B

Use the magento-cloud route:add command to create a route to the Staging URL

☐ C

Modify the MAGENTO_CLOUD_ROUTES environment variable on your staging environment

☐ D

Configure UPDATE_URLS to true in the .magento.env.yaml file

You are reviewing a third party extension for Magento Commerce Cloud compatibility. You discover the
module writes to a custom directory called internal/ within the Magento base directory.
The directory has to be shared by all web servers.
What do you do so the module can be used on Magento Commerce Cloud?

☑ A

Declare a shared mount for the internal/ directory in .magento.app.yaml

☐ B

Request the vendor make the directory configurable in the Magento admin

☐ C

Add a patch for extension so it uses the subdirectory var/internal/ instead

☐ D

Create a symlink from /tmp/internal to the internal/ folder

You need to disable a module on a Magento Commerce 2.3 Cloud project and remove its database tables.
The module uses the declarative schema system to manage its database changes.
Which action do you take?

☑ A

Delete the module from the git repository leaving the record in app/etc/config.php intact and deploy the changes

☐ B

Remove the module line from the app/etc/config.php file on the local environment and then deploy the file

☐ C

Run bin/magento module:disable MyCompany_MyModule on the production environment and download and commit the app/etc/config.php file

☐ D

Run bin/magento module:disable MyCompany_MyModule on the local environment and then commit and deploy the app/eta/config.php file

Your client is expecting a five-day sale where the traffic will approximately double. Your instance is consuming all available resources on all three nodes so upsizing one level is appropriate.
How many upsize days will this event use?

A

Fifteen

B

Ten

C

Eight

☑ **D**

Five

After contacting Magento Support, you were advised to permanently enable the Redis disable_locking option.
How do you implement this on a Production environment?

☑ **A**

In app/etc/env.php, add disable_locking: 1 inside the Redis session configuration block.

B

Add the SESSION_CONFIGURATION environment variable into .magento.env.yaml with the correct value.

C

Add disable_locking = true into the php.ini file in the Magento project root.

D

Modify the .magento.app.yaml file to add the REDIS_SESSION_CONFIGURATION environment variable.

You add a new Composer dependency utilizing composer require. After testing composer install locally, you add the Composer lock and json files and perform a deployment. The build phase fails as

Composer is unable to resolve the dependencies.
Why did this happen?

☐ A

You did not execute the install locally with --require-dev

☐ B

The project cache should have been cleared with the magento-cloud:project-build-cache command

☐ C

Your commit needs to contain the updated vendor folder

☑ D

The versions of PHP on your local environment and integration environment differ

Your project is in a phase where the locales that will be used are still being determined. You need to configure your project in a manner that will allow them to be changed dynamically.
How can you accomplish this?

☐ **A**

Remove the system/default/general/locale configuration from the app/etc/config.php file

☐ **B**

Configure DYNAMIC_LOCALES in the SCD_MATRIX to true in the .magento.env.yaml file

☑ **C**

Configure SCD_ON_DEMAND to true in the .magento.env.yaml file

☐ **D**

Run the command vendor/bin/ece-tools locales:unlock on the environment

Exploring the Production environment, you find MYSQL_USE_SLAVE_CONNECTION is enabled.
What is the purpose of this setting?

☐ **A**

To enable fault tolerance on the MySQL cluster.

☐ **B**

To allow database backups from a MySQL slave server with minimal impact on the Production site.

☑ **C**

To utilize the read-only slave servers and improve the overall performance.

☐ **D**

To split the database tables between the three Production servers and improve performance.

A project has a Magento cron job implemented to export new order data to a file. This process can also be
triggered by a custom bin/magento command. The functionality uses a lock file in var/locks to prevent
concurrent execution.
There are instances when the process gets locked after Production deployments and order data is not exported.
How do you prevent this from happening?

☐ **A**

You must access the Production environment and delete the lock file

☐ **B**

Use the VAR_LOCKS_CLEAR variable available for the build stage in the .magento.app.yaml file

☐ **C**

Write a custom post_deploy hook to delete orphan lock files and include it in the .magento.app.yaml file

☑ **D**

Use the VAR_LOCKS_CLEAR variable available for the post_deploy stage in the .magento.env.yaml file

Magento Support advises you to upgrade to the latest release of ece-tools matching the project's Magento
Commerce version 2.3.1.
How do you do that?

☑ A

Run the command composer update magento/ece-tools

☐ B

Require the exact version with composer require magento/ece-tools:2.3.1-pX where X is the latest release number

☐ C

Run the command ece-tools self-upgrade

☐ D

Clone the repository github.com/magento/ece-tools and copy the src/ folder to vendor/ magento/ece-tools/src

Magento Commerce Cloud comes with a variety of additional features that sets it apart from the on-premise
Magento Commerce and Magento Open Source platforms.
What feature improves the release update process and simplifies commands to create a backup of the
database, apply custom patches, and verify environment configuration?

☐ A

Docker environment

☐ B

Magento Cloud-specific CLI

☐ C

Cloud configuration wizards

☑ D

ECE-Tools package